CLAIMS

What is claimed is:

1. An apparatus comprising:

a processor having a normal execution mode and a host execution mode; and

a virtual machine monitor (VMM) operable in conjunction with the host execution mode to create at least one protected mode environment to operate guest software in a virtual machine;

wherein responsive to a command to switch between protected modes, the VMM causes the processor to atomically switch between an original protected mode environment and a target protected mode environment.

2. The apparatus of claim 1, wherein switching between protected modes further includes entering a virtual machine execution (VMX) mode to enable virtual machine functionality.

3. The apparatus of claim 1, further comprising a virtual machine control structure (VMCS) to store state information for use in switching between the original protected mode environment and the target protected mode environment, the VMCS to store state information related to the original protected mode environment.

4. The apparatus of claim 3, wherein the virtual machine control structure (VMCS) further stores state information related to the target protected mode environment.

5. The apparatus of claim 4, wherein the virtual machine control structure (VMCS) further stores a guest entry point field to point to a command used for instructing the processor to exit out of the original protected mode environment and a host entry point field to point to a command to instruct the processor to exit out of a virtual machine execution (VMX) mode.

6. The apparatus of claim 1, wherein the VMM causes the processor to enter a virtual machine execution (VMX) mode, to exit out of the original protected mode environment, and to enter into the target protected mode environment.

7. The apparatus of claim 6, wherein the VMM causes the target protected mode environment to exit out of the virtual machine (VMX) extension mode.

8. The apparatus of claim 7, wherein the processor resumes operation with the target protected mode environment.

9. The apparatus of claim 1, wherein guest software operable in a protected mode environment includes an operating system.

10. A method comprising:

providing a normal execution mode in a processor and a host execution mode in a processor;

creating at least one protected mode environment to operate guest software in a virtual machine; and

wherein responsive to a command to switch between protected modes, atomically switching between an original protected mode environment and a target protected mode environment.

11. The method of claim 10, wherein switching between protected modes further includes entering a virtual machine execution (VMX) mode to enable virtual machine functionality.

12. The method of claim 10, further comprising storing state information for use in switching between the original protected mode environment and the target protected mode environment including storing state information related to the original protected mode environment.

13. The method of claim 12, further comprising storing state information related to the target protected mode environment.

14. The method of claim 13, further comprising:

storing a guest entry point field to point to a command used for instructing the processor to exit out of the original protected mode environment; and

storing a host entry point field to point to a command to instruct the processor to exit out of a virtual machine execution (VMX) mode.

15. The method of claim 10, further comprising
entering a virtual machine execution (VMX) mode;
exiting out of the original protected mode environment; and
entering into the target protected mode environment.

16. The method of claim 15, further comprising exiting out of the virtual machine (VMX) extension mode.

17. The method of claim 16, further comprising resuming operation with the target protected mode environment.

18. The method of claim 10, wherein guest software operable in a protected mode environment includes an operating system.

19. A machine-readable medium having stored thereon instructions, which when executed by a machine, cause the machine to perform the following operations comprising:
providing a normal execution mode in a processor and a host execution mode in a processor;
creating at least one protected mode environment to operate guest software in a virtual machine; and
wherein responsive to a command to switch between protected modes, atomically switching between an original protected mode environment and a target protected mode environment.

20. The machine-readable medium of claim 19, wherein switching between protected modes further includes entering a virtual machine execution (VMX) mode to enable virtual machine functionality.

21. The machine-readable medium of claim 21, further comprising storing state information for use in switching between the original protected mode environment

and the target protected mode environment including storing state information related to the original protected mode environment.

22. The machine-readable medium of claim 21, further comprising storing state information related to the target protected mode environment.

23. The machine-readable medium of claim 22, further comprising:

storing a guest entry point field to point to a command used for instructing the processor to exit out of the original protected mode environment; and

storing a host entry point field to point to a command to instruct the processor to exit out of a virtual machine execution (VMX) mode.

24. The machine-readable medium of claim 19, further comprising

entering a virtual machine execution (VMX) mode;

exiting out of the original protected mode environment; and

entering into the target protected mode environment.

25. The machine-readable medium of claim 24, further comprising exiting out of the virtual machine (VMX) extension mode.

26. The machine-readable medium of claim 25, further comprising resuming operation with the target protected mode environment.

27. The machine-readable medium of claim 19, wherein guest software operable in a protected mode environment includes an operating system.

28. A system comprising:

a processor including virtual machine extension (VMX) instruction support, the processor further having a normal execution mode and a host execution mode; and

a virtual machine monitor (VMM) operable in conjunction with the host execution mode to create at least one protected mode environment to operate guest software in a protected memory area;

wherein responsive to a command to switch between protected modes, the VMM causes the processor to atomically switch between an original protected mode environment and a target protected mode environment.

29. The system of claim 28, wherein switching between protected modes further includes entering a virtual machine execution (VMX) mode to enable virtual machine functionality.

30. The system of claim 28, further comprising a virtual machine control structure (VMCS) to store state information for use in switching between the original protected mode environment and the target protected mode environment, the VMCS to store state information related to the original protected mode environment.

31. The system of claim 30, wherein the virtual machine control structure (VMCS) further stores state information related to the target protected mode environment.

32. The system of claim 31, wherein the virtual machine control structure (VMCS) further stores a guest entry point field to point to a command used for instructing the processor to exit out of the original protected mode environment and a host entry point field to point to a command to instruct the processor to exit out of a virtual machine execution (VMX) mode.

33. The system of claim 28, wherein the VMM causes the processor to enter a virtual machine execution (VMX) mode, to exit out of the original protected mode environment, and to enter into the target protected mode environment.

34. The system of claim 33, wherein the VMM causes the target protected mode environment to exit out of the virtual machine (VMX) extension mode.

35. The system of claim 34, wherein the processor resumes operation with the target protected mode environment.

36. The system of claim 28, wherein guest software operable in a protected mode environment includes an operating system.